

REMARKS

Claims 1-7 were pending in the Application. Claims 21-33 are added. Hence, claims 1-7 and 21-33 are pending.

Claims 1-7 are rejected under 35 U.S.C. §112, second paragraph. Claims 1-7 are rejected under 35 U.S.C. §103(a). Applicants address these rejections below.

Applicants thank Examiner Lee for discussing the Office Action with Applicants' Attorney, Bobby Voigt, on February 12, 2009.

I. REJECTIONS UNDER 35 U.S.C. §112, SECOND PARAGRAPH:

The Examiner has rejected claims 1-7 under 35 U.S.C. §112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which Applicants regard as the invention. Office Action (12/22/2008), page 3. In particular, the Examiner asserts that "said first deployable software component" and "said second deployable software component" in claim 1 needs to be identified in either "a first descriptor file" or a "second descriptor file." *Id.* However, as Applicants' Attorney, Bobby Voigt, explained to Examiner Lee on February 12, 2009, claim 1 states that the interfaces for a deployable software component are compared in the first descriptor file and in the second descriptor file. As discussed in Applicants' Specification, the interfaces for the current Enterprise JavaBeans in the input and output Java Archive (JAR) files are compared. Specification, page 8, lines 24-25. If one or more of these miscompare, then the current Bean is tagged. Specification, page 8, lines 25-26. Hence, different deployable software components are not being compared in different descriptor files. Thus, it would not make sense to add the phrase "in a first descriptor file" or "in a second descriptor file" after the phrase "said first deployable software component" or "said second deployable software component" as requested by the Examiner.

Thus, the scope of claims 1-7 is clear to a hypothetical person possessing the ordinary level of skill in the pertinent art. According to M.P.E.P. §2171, a claim particularly points out and distinctly defines the metes and bounds of the subject matter if the scope of the claim is clear to a hypothetical person possessing the ordinary level of skill in the pertinent art. Hence, claims 1-7 particularly point out and distinctly define the metes and bounds of the subject matter. Consequently, Applicants respectfully assert that claims 1-7 are allowable under 35 U.S.C. §112, second paragraph, and respectfully request the Examiner to withdraw the rejections of claims 1-7 under 35 U.S.C. §112, second paragraph. M.P.E.P. §2171.

II. REJECTIONS UNDER 35 U.S.C. §103(a):

The Examiner has rejected claims 1, 2, 4 and 7 under 35 U.S.C. §103(a) as being unpatentable over Garms et al. (U.S. Patent No. 7,296,255) (hereinafter "Garms") in view of Spring et al. (U.S. Patent No. 6,971,093) (hereinafter "Springs"). The Examiner has further rejected claim 3 under 35 U.S.C. §103(a) as being unpatentable over Garms in view of Spring and in further view of Cohen et al. (U.S. Patent Application Publication No. 2004/0034849) (hereinafter "Cohen"). Additionally, the Examiner has rejected claims 5 and 6 under 35 U.S.C. §103(a) as being unpatentable over Garms in view of Spring and in further view of Kovacs et al. (U.S. Patent Application Publication No. 2004/0158571) (hereinafter "Kovacs"). Applicants respectfully traverse these rejections for at least the reasons stated below and respectfully request that the Examiner reconsider and withdraw these rejections.

A. Claims 1, 2, 4, 7, 21-22, 24, 27-29 and 31 are not properly rejected under 35 U.S.C. §103(a) as being unpatentable over Garms in view of Spring.

1. Claims 1, 21 and 28 are patentable over Garms in view of Spring.

Applicants respectfully assert that Garms and Spring, taken singly or in combination, do not teach "for each deployable software component in an preselected input archive file, comparing interfaces for the deployable software component identified in a first descriptor file in said input archive file and a second descriptor file in a preselected output archive file" as recited in claim 1 and similarly in claims 21 and 28. The Examiner cites column 4, lines 63-67 and column 5, lines 1-12 of Garms as well as cites column 9, lines 38-46; column 10, lines 19-26 and column 13, lines 49-63 of Spring as teaching the above-cited claim limitations. Office Action (12/22/2008), pages 4 and 6. Applicants respectfully traverse.

Garms teaches examining at least one deployment descriptor for the application to determine if the at least one deployment descriptor contains an entry for each of the at least one deployable module 202. Column 5, lines 1-4. Garms further teaches that if there were deployable modules identified without corresponding entries in the application's deployment descriptors, then a list of modules to deploy containing each deployable module that was not found in the deployment descriptors is created. Column 5, lines 5-10.

Hence, Garms teaches creating a list of modules to deploy for those deployable modules not found in the deployment descriptors.

Spring teaches using a current data structure describing the interface for the newest version of the core module and comparing the current data structure to the previous data structure describing the interface for the previous version of the core module. Column 9, lines 39-43. Spring further teaches that if the contents of the two data structures differ, the minor version number is incremented by one. Column 9, lines 45-46.

Hence, Spring teaches comparing the current data structure to the previous data structure and if the contents of the two data structures differ, the minor version number is incremented by one.

There is no language in the cited passages that teaches comparing interfaces for the deployable software component identified in a first descriptor file in the input archive file and a second descriptor file in a preselected output archive file. Instead, Spring teaches comparing the current data structure to the previous data structure describing the interface for the previous version of the core module. Spring does not teach comparing the interfaces identified in a first descriptor file or a second descriptor file. Neither is there any language in the cited passages that teaches comparing interfaces for the deployable software component identified in a first descriptor file in the input archive file and a second descriptor file in a preselected output archive file for each deployable software component in a preselected input archive file. Garms teaches creating a list of modules to deploy for those deployable modules not found in the deployment descriptors. However, there is no language in either Garms or Springs that teaches performing the comparison step for each deployable software component.

Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 1, 21 and 28, since the Examiner is relying upon incorrect, factual predicates in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

Applicants further assert that Garms and Spring, taken singly or in combination, do not teach "if the comparing step miscompares for a first deployable software component, tagging said first deployable software component" as recited in claim 1 and similarly in claims 21 and 28. The Examiner cites column 5, lines 1-12 of Garms as teaching the above-cited claim limitation. Office Action (12/22/2008), page 5. Applicants respectfully traverse.

As stated above, Garms teaches creating a list of modules to deploy for those deployable modules not found in the deployment descriptors.

There is no language in the cited passage that teaches tagging a first deployable software component if the comparing step miscompares for the first deployable software component. There is no language in Garms that teaches the comparing step, namely, comparing between the interfaces for the deployable software component in the input archive file and in the output archive file. Instead, Garms teaches determining if deployable modules are identified without corresponding entries in the application's deployment descriptors.

Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 1, 21 and 28, since the Examiner is relying upon incorrect, factual predicates in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

Applicants further assert that Garms and Spring, taken singly or in combination, do not teach "if the comparing step miscompares for a second deployable software component, tagging said second deployable software component" as recited in claim 1 and similarly in claims 21 and 28. The Examiner cites column 5, lines 1-12 of Garms as teaching the above-cited claim limitation. Office Action (12/22/2008), page 5. Applicants respectfully traverse.

As stated above, Garms teaches creating a list of modules to deploy for those deployable modules not found in the deployment descriptors.

There is no language in the cited passage that teaches tagging a second deployable software component if the comparing step miscompares for the second deployable software component. There is no language in Garms that teaches the comparing step, namely, comparing between the interfaces for the deployable software component in the input archive file and in the output archive file. Instead, Garms teaches determining if

deployable modules are identified without corresponding entries in the application's deployment descriptors.

Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 1, 21 and 28, since the Examiner is relying upon incorrect, factual predicates in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

- b. Claims 2, 4 and 7 are patentable over Garms in view of Spring for at least the above-stated reasons that claim 1 is patentable over Garms in view of Spring.

Claims 2, 4 and 7 each recite combinations of features of independent claim 1, and hence claims 2, 4 and 7 are patentable over Garms in view of Spring for at least the above-stated reasons that claim 1 is patentable over Garms in view of Spring.

- c. Claims 22, 24 and 27 are patentable over Garms in view of Spring for at least the above-stated reasons that claim 21 is patentable over Garms in view of Spring.

Claims 22, 24 and 27 each recite combinations of features of independent claim 21, and hence claims 22, 24 and 27 are patentable over Garms in view of Spring for at least the above-stated reasons that claim 21 is patentable over Garms in view of Spring.

- d. Claims 29 and 31 are patentable over Garms in view of Spring for at least the above-stated reasons that claim 28 is patentable over Garms in view of Spring.

Claims 29 and 31 each recite combinations of features of independent claim 28, and hence claims 29 and 31 are patentable over Garms in view of Spring for at least the above-stated reasons that claim 28 is patentable over Garms in view of Spring.

- e. Claims 2, 22 and 29 are patentable over Garms in view of Spring.

Applicants respectfully assert that Garms and Springs, taken singly or in combination, do not teach "wherein tagging a deployable software component comprises

storing a name of the deployable software component in a file" as recited in claim 2 and similarly in claims 22 and 29. The Examiner cites column 6, lines 19-65 of Garms as teaching the above-cited claim limitation. Office Action (12/22/2008), page 6. Applicants respectfully traverse.

Garms instead shows that exemplary deployment descriptors might be automatically created and updated by the iterative deployment process. Column 6, lines 18-20. Garms further teaches that these examples are taken from a J2EE environment and include application.xml and weblogic-application.xml. Column 6, lines 20-23.

There is no language in the cited passage that teaches that tagging a deployable software component comprises storing a name of the deployable software component in a file.

Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 2, 22 and 29, since the Examiner is relying upon incorrect, factual predicates in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

f. Claims 4, 24 and 31 are patentable over Garms in view of Spring.

Applicants respectfully assert that Garms and Springs, taken singly or in combination, do not teach "if the first descriptor file and second descriptor file compare for the first deployable software component, introspecting a binary class file for the first deployable software component in the input and output archive files; and if, in response to the introspection, a signature or return type of an interface of said binary class files miscompare, tagging the first deployable software component" as recited in claim 4 and similarly in claims 24 and 31. The Examiner cites column 5, lines 1-12 of Garms and further cites column 13, lines 1-24 of Spring as teaching the above-cited claim limitations. Office Action (12/22/2008), page 7. Applicants respectfully traverse.

As stated above, Garms teaches creating a list of modules to deploy for those deployable modules not found in the deployment descriptors.

Spring teaches that the data structure can be used to describe a software interface with any programming language that has typed declarations for fields, such as constants and variables, and routines. Column 13, lines 4-7.

There is no language in the cited passages that teaches introspecting a binary class file for the first deployable software component in the input and output archive files. Neither is there any language in the cited passages that discloses introspecting a binary class file for the first deployable software component in the input and output archive files if the first descriptor file and second descriptor file compare for the first deployable software component. Neither is there any language in the cited passages that discloses tagging the first deployable software component if, in response to the introspection, a signature or return type of an interface of the binary class files miscompare.

Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 4, 24 and 31, since the Examiner is relying upon incorrect, factual predicates in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

- g. Claims 7 and 27 are patentable over Garms in view of Spring.

Applicants respectfully assert that Garms and Springs, taken singly or in combination, do not teach "wherein the comparing, tagging and deploying steps are performed in response to an execution of a build script invoking a selective deployer utility" as recited in claim 7 and similarly in claim 27. The Examiner cites column 3, lines 36-59 of Garms as teaching the above-cited claim limitation. Office Action (12/22/2008), page 7. Applicants respectfully traverse.

Garms instead teaches that developing these deployment descriptors manually is tedious and time consuming. Column 3, lines 36-37. Garms further teaches that each time the deployment configuration changes, the deployment descriptors must change to match. Column 3, lines 36-38. Additionally, Garms teaches that using incremental application deployment, all the files under development can be modified directly in place on the server's disk. Column 3, lines 46-48. Furthermore, Garms teaches that application and module configuration information can be collected from the user as they build their application, e.g. using an Integrated Development Environment (IDE). Column 3, lines 48-51.

There is no language in the cited passage that teaches that the comparing, tagging and deploying steps are performed in response to an execution of a build script. Neither is there any language in the cited passage that teaches that the comparing, tagging and deploying steps are performed in response to an execution of a build script invoking a selective deployer utility.

Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 7 and 27, since the Examiner is relying upon incorrect, factual predicates in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

2. Examiner's reasoning for modifying Garms with Spring to include the missing claim limitation of claims 1, 21 and 28 is insufficient to establish a *prima facie* case of obviousness.

Most if not all inventions arise from a combination of old elements. *See In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Obviousness is determined from the vantage point of a hypothetical person having ordinary skill in the art to which the patent pertains. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Therefore, an Examiner may often find every element of a claimed invention in the prior art. *Id.* However, identification in the prior art of each individual part claimed is insufficient to defeat patentability of the whole claimed invention. *See Id.* In order to establish a *prima*

facie case of obviousness, the Examiner must show reasons that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would select the elements from the cited prior art references for combination in the manner claimed. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998). The Examiner must provide articulated reasoning with some rational underpinning to support the legal conclusion of obviousness. *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006) (cited approvingly in *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007)).

The Examiner admits that Garms does not teach the aspect of comparing interfaces for the deployable software component identified in a first descriptor file and a second descriptor file, as recited in claim 1 and similarly in claims 21 and 28. Office Action (12/22/2008), page 6. The Examiner asserts that Spring teaches the above-cited claim limitation. *Id.* The Examiner's reasoning for modifying Garms with Spring to include the above-cited claim limitation is "for further optimizing compatibility (e.g., version compatibility) between the construct modules list and the application deployment descriptor as taught in Spring (e.g., col. 3: 30-44)." *Id.* The Examiner's reasoning is insufficient to establish a *prima facie* case of obviousness in rejecting claims 1-7 and 21-33.

As stated above, the Examiner cites column 3, lines 30-44 of Spring as support for the Examiner's reasoning for modifying Garms with Spring to include the above-cited missing claim limitation of claims 1, 21 and 28. Spring teaches that determining the earliest version number of the interacting module using the past manual approach involves testing the module with several earlier versions until one is found that is not compatible or involves examining all the features of the interface. Column 3, lines 30-36. Spring additionally teaches that this is a tedious process and subject to more human errors. Column 3, lines 36-37. Hence, Spring teaches that the manual techniques in

determining the earliest version number of the interacting module are tedious and subject to error.

There is no language in Spring (and in particular column 3, lines 30-44) that makes any suggestion to compare interfaces for the deployable software component identified in a first descriptor file and a second descriptor file (missing claim limitation) in order to optimizing compatibility between the construct modules list and the application deployment descriptor. The Examiner has to provide some rational connection between the cited passage that is the source of the Examiner's reasoning and the above-cited missing claim limitation. The Examiner's source of reasoning (column 3, lines 30-44) does not provide reasons as to why one skilled in the art would modify Garms to include the above-cited missing claim limitation of claims 1, 21 and 28. Accordingly, the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 1-7 and 21-33. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007).

Further, the Examiner's reasoning ("further optimizing compatibility (e.g., version compatibility) between the construct modules list and the application deployment descriptor") does not provide reasons, as discussed further below, that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would modify Garms to include the above-indicated missing claim limitation of claims 1, 21 and 28. Accordingly, the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 1-7 and 21-33. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007); *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998).

Garms addresses the problems in connection with manually developing deployment descriptors. Column 3, lines 36-40. The Examiner has not provided any reasons as to why one skilled in the art would modify Garms (which addresses the

problems in connection with manually developing deployment descriptors) to compare interfaces for the deployable software component identified in a first descriptor file and a second descriptor file (missing claim limitation). The Examiner's rationale ("further optimizing compatibility (e.g., version compatibility) between the construct modules list and the application deployment descriptor") does not provide such reasoning.

Why would the reason to modify Garms (whose purpose is to address the problems in connection with manually developing deployment descriptors) to compare interfaces for the deployable software component identified in a first descriptor file and a second descriptor file (missing claim limitation) be to optimize compatibility (e.g., version compatibility) between the construct modules list and the application deployment descriptor? Garms is not concerned with optimizing compatibility (e.g., version compatibility) between the construct modules list and the application deployment descriptor. The Examiner cannot completely ignore the teachings of Garms in concluding it would have been obvious to modify Garms to include the above-cited missing claim limitation of claims 1, 21 and 28.¹

¹ For example, suppose that the invention of a super soaker gun (essentially a plastic gun that shoots water) was never developed and an Applicant filed for a patent application on the super soaker gun. Applicant claims a plastic gun with a container of water that shoots water. The Examiner cites a primary reference that teaches a plastic gun that shoots darts and cites a secondary reference that teaches a plastic toy that contains a container of water. Since the primary reference does not teach a container filled with water, the Examiner cites the secondary reference as teaching this missing claim limitation. The secondary reference specifically states that the purpose of the container is to carry water. The Examiner then concludes that it would have been obvious to modify the primary reference with the secondary reference in order to carry water. The Examiner believes that he/she has established a *prima facie* case of obviousness since the Examiner has found a reason to have a container of water. However, the Examiner is completely ignoring the teaching of the primary reference. Why would one skilled in the art modify a plastic gun that shoots darts to have a container of water? This is the key question to answer. While having a container of water may be used to carry water, that is irrelevant as far as the purpose of the primary reference. Simply citing to a passage in the secondary reference that discusses the purpose of that secondary reference may not be sufficient evidence for an obviousness rejection. After all, surely there is a reason as to why the secondary reference teaches the missing claim limitation or else why would the secondary reference include it? The Examiner must explain the connection between the teachings of the primary reference and the rationale of the secondary reference for including the missing claim limitation. Otherwise, everything can be deemed obvious and virtually nothing can be patented.

Further, what is the rational connection between comparing interfaces for the deployable software component identified in a first descriptor file and a second descriptor file (missing claim limitation) and optimizing compatibility (e.g., version compatibility) between the construct modules list and the application deployment descriptor (Examiner's reasoning)?

Hence, the Examiner's rationale does not provide reasons that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would modify Garms to include the above-cited missing claim limitation of claims 1, 21 and 28. Accordingly, the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 1-7 and 21-33. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007); *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998).

3. Examiner fails to provide a rational underpinning for modifying Garms with Spring to include the missing claim limitations of claims 4, 24 and 31.

As stated above, most if not all inventions arise from a combination of old elements. *See In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Obviousness is determined from the vantage point of a hypothetical person having ordinary skill in the art to which the patent pertains. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Therefore, an Examiner may often find every element of a claimed invention in the prior art. *Id.* However, identification in the prior art of each individual part claimed is insufficient to defeat patentability of the whole claimed invention. *See Id.* In order to establish a *prima facie* case of obviousness, the Examiner must show reasons that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would select the elements from the cited prior art references for combination in the manner claimed. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998). The Examiner must provide articulated reasoning with some

rational underpinning to support the legal conclusion of obviousness. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007).

As understood by Applicants, the Examiner admits that Garms does not teach all of the claim limitations of claims 4, 24 and 31. Office Action (12/22/2008), page 7. The Examiner asserts that Spring teaches these missing claim limitations of claims 4, 24 and 31. *Id.* However, the Examiner has not provided any rational underpinning for modifying Garms with Spring to include these missing claim limitations. Hence, the Examiner has not provided a *prima facie* case of obviousness in rejecting claims 4, 24 and 31. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007); M.P.E.P. §2143.

- B. Claims 3, 23 and 30 are not properly rejected under 35 U.S.C. §103(a) as being unpatentable over Garms in view of Spring and in further view of Cohen.
 - 1. Garms, Spring and Cohen, taken singly or in combination, do not teach at least the following claim limitations.

Applicants respectfully assert that Garms, Spring and Cohen, taken singly or in combination, do not teach "if the first descriptor file and second descriptor file compare for the first deployable software component, comparing a size of a binary class file for the first deployable software component in the input and output archive files; and if the size of said binary class files miscompare, tagging the first deployable software component" as recited in claim 3 and similarly in claims 23 and 30. The Examiner cites column 5, lines 1-12 of Garms and paragraph [0091] of Cohen as teaching the above-cited claim limitations. Office Action (12/22/2008), pages 8-9. Applicants respectfully traverse.

As stated above, Garms teaches creating a list of modules to deploy for those deployable modules not found in the deployment descriptors.

Cohen teaches that the size of the compressed delta binary file is compared to the compressed delta binary file data B. [0091].

Hence, Cohen teaching comparing the size of a compressed binary file with another compressed binary file.

There is no language in the cited passages that teaches comparing a size of a binary class file for the first deployable software component in the input and output archive files. Neither is there any language in the cited passages that teaches comparing a size of a binary class file for the first deployable software component in the input and output archive files if the first descriptor file and second descriptor file compare for the first deployable software component. Neither is there any language in the cited passages that teaches tagging the first deployable software component. Neither is there any language in the cited passages that teaches tagging the first deployable software component if the size of the binary class files miscompare.

Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 3, 23 and 30 since the Examiner is relying upon incorrect, factual predicates in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

2. Examiner's reasoning for modifying Garms with Cohen to include the missing claim limitations of claims 3, 23 and 30 is insufficient to establish a *prima facie* case of obviousness.

As stated above, most if not all inventions arise from a combination of old elements. *See In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Obviousness is determined from the vantage point of a hypothetical person having ordinary skill in the art to which the patent pertains. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Therefore, an Examiner may often find every element of a claimed invention in the prior art. *Id.* However, identification in the prior art of each individual part claimed is insufficient to defeat patentability of the whole claimed invention. *See Id.* In order to

establish a *prima facie* case of obviousness, the Examiner must show reasons that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would select the elements from the cited prior art references for combination in the manner claimed. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998). The Examiner must provide articulated reasoning with some rational underpinning to support the legal conclusion of obviousness. *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006) (cited approvingly in *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007)).

The Examiner admits that Garms does not teach "comparing a size of a binary class file for the first deployable software component " as recited in claim 3 and similarly in claims 23 and 30. Office Action (12/22/2008), page 8. The Examiner asserts that Cohen teaches the above-cited claim limitation. *Id.* at pages 8-9. The Examiner's reasoning for modifying Garms with Cohen to include the above-cited claim limitation is "for further identifying differences of the deployable module between the construct modules list and the application deployment descriptor as taught in Cohen (e.g., [0091]."

Id. at page 9. The Examiner's reasoning is insufficient to establish a *prima facie* case of obviousness in rejecting claims 3, 23 and 30.

As stated above, the Examiner cites paragraph [0091] of Cohen as support for the Examiner's reasoning for modifying Garms with Cohen to include the above-cited missing claim limitation of claims 3, 23 and 30. Cohen teaches comparing the size of a compressed binary file with another compressed binary file. [0091].

There is no language in Cohen (and in particular paragraph [0091]) that makes any suggestion to compare a size of a binary class file for the first deployable software component (missing claim limitation) in order for identifying differences of the deployable module between the construct modules list and the application deployment descriptor (Examiner's reasoning). The Examiner has to provide some rational

connection between the cited passage that is the source of the Examiner's reasoning and the above-cited missing claim limitation. The Examiner's source of reasoning (paragraph [0091] of Cohen) does not provide reasons as to why one skilled in the art would modify Garms to include the above-cited missing claim limitation of claims 3, 23 and 30. Accordingly, the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 3, 23 and 30. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007).

Further, the Examiner's reasoning ("for identifying differences of the deployable module between the construct modules list and the application deployment descriptor") does not provide reasons, as discussed further below, that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would modify Garms to include the above-indicated missing claim limitation of claims 3, 23 and 30. Accordingly, the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 3, 23 and 30. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007); *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998).

Garms addresses the problem of long delays in the development cycle in developing software. Column 1, line 30 – column 3, line 40. The Examiner has not provided any reasons as to why one skilled in the art would modify Garms (which addresses the problem of long delays in the development cycle in developing software) to compare a size of a binary class file for the first deployable software component (missing claim limitation). The Examiner's rationale ("for identifying differences of the deployable module between the construct modules list and the application deployment descriptor") does not provide such reasoning.

Why would the reason to modify Garms (whose purpose is to address the problem of long delays in the development cycle in developing software) to compare a size of a

binary class file for the first deployable software component (missing claim limitation) be to identify differences of the deployable module between the construct modules list and the application deployment descriptor?

What is the rational connection between comparing a size of a binary class file for the first deployable software component (missing claim limitation) and identifying differences of the deployable module between the construct modules list and the application deployment descriptor?

Hence, the Examiner's rationale does not provide reasons that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would modify Garms to include the above-cited missing claim limitation of claims 3, 23 and 30. Accordingly, the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 3, 23 and 30. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007); *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998).

- C. Claims 5, 6, 25, 26, 32 and 33 are not properly rejected under 35 U.S.C. §103(a) as being unpatentable over Garms in view of Spring and Kovacs.
 - 1. Garms and Kovacs, taken singly or in combination, do not teach at least the following claim limitations.
 - a. Claims 5, 25 and 32 are patentable over Garms in view of Spring and Kovacs.

Applicants respectfully assert that Garms, Spring and Kovacs, taken singly or in combination, do not teach "opening said preselected output archive file; and if the step of opening the preselected output archive fails, tagging each deployable software component in the input archive file" as recited in claim 5. The Examiner cites element 302 in Figure 3 and paragraphs [0020-0021] of Kovacs as teaching the above-cited claim limitations. Office Action (12/22/2008), pages 9-10 Applicants respectfully traverse.

Kovacs instead teaches that the builder invokes validator 302 to locate errors within deployment descriptor files (e.g., incorrect CMP field name, etc.). [0020]. Kovacs further teaches that when compiler 304 determines that there is an error in a deployment descriptor file, it can create an Error object to store an error code and/or message, the identifier of a node and corresponding field in the resource hierarchy 102 to which the error pertains, and/or an XML type of the field. [0020]. Kovacs further teaches that validator 302 and/or compiler 304 can display human-readable error messages corresponding to each error object in message area 106. [0021].

Hence, Kovacs teaches that when the compiler determines that there is an error in an deployment descriptor file, it can create an error object to store an error code and/or message.

There is no language in the cited passages that teaches opening the preselected output archive file. Neither is there any language in the cited passages that teaches tagging each deployable software component in the input archive file. Neither is there any language in the cited passages that teaches tagging each deployable software component in the input archive file if the step of opening the preselected output archive fails.

Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 5, 25 and 32, since the Examiner is relying upon incorrect, factual predicates in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

- b. Claims 6, 26 and 33 are patentable over Garms in view of Kovacs.

Applicants respectfully assert that Garms, Spring and Kovacs, taken singly or in combination, do not teach "wherein the step of tagging each deployable software component is performed in response to the step of opening the preselected output archive throwing an exception" as recited in claim 6 and similarly in claims 26 and 33. The

Examiner cites paragraphs [0020-0021] of Kovacs as teaching the above-cited claim limitations. Office Action (12/22/2008), page 10. Applicants respectfully traverse.

As stated above, Kovacs instead teaches that the builder invokes validator 302 to locate errors within deployment descriptor files (e.g., incorrect CMP field name, etc.). [0020]. Kovacs further teaches that when compiler 304 determines that there is an error in an deployment descriptor file, it can create an Error object to store an error code and/or message, the identifier of a node and corresponding field in the resource hierarchy 102 to which the error pertains, and/or an XML type of the field. [0020]. Kovacs further teaches that validator 302 and/or compiler 304 can display human-readable error messages corresponding to each error object in message area 106. [0021].

There is no language in the cited passages that teaches that the step of tagging each deployable software component is performed in response to the step of opening the preselected output archive throwing an exception. Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 6, 26 and 33, since the Examiner is relying upon incorrect, factual predicates in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

2. Examiner's reasoning for modifying Garms with Kovacs to include the missing claim limitations of claims 5, 25 and 32 is insufficient to establish a *prima facie* case of obviousness.

As stated above, most if not all inventions arise from a combination of old elements. *See In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Obviousness is determined from the vantage point of a hypothetical person having ordinary skill in the art to which the patent pertains. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Therefore, an Examiner may often find every element of a claimed invention in the prior art. *Id.* However, identification in the prior art of each individual part claimed is insufficient to defeat patentability of the whole claimed invention. *See Id.* In order to establish a *prima facie* case of obviousness, the Examiner must show reasons that the

skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would select the elements from the cited prior art references for combination in the manner claimed. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998). The Examiner must provide articulated reasoning with some rational underpinning to support the legal conclusion of obviousness. *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006) (cited approvingly in *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007)).

As understood by Applicants, the Examiner admits that Garms does not teach "opening said preselected output archive file; and if the step of opening the preselected output archive fails, tagging each deployable software component in the input archive file" as recited in claim 5 and similarly in claims 25 and 32. Office Action (12/22/2008), page 9. The Examiner asserts that Kovacs teaches the above-cited claim limitations. *Id.* at pages 9-10. The Examiner's reasoning for modifying Garms with Kovacs to include the above-cited claim limitations is "to identify the input errors and offer the suggesting solution to those errors (see Kovacs, page 2, [0021])." *Id.* at page 10. The Examiner's reasoning is insufficient to establish a *prima facie* case of obviousness in rejecting claims 5-6, 25-26 and 32-33.

As stated above, the Examiner cites paragraph [0021] of Kovacs as support for the Examiner's reasoning for modifying Garms with Kovacs to include the above-cited missing claim limitations of claims 5, 25 and 32. Kovacs teaches that validator 302 and/or compiler 304 can display human-readable error messages corresponding to each error object in message area 106. [0021]. Kovacs further teaches that validator 302 can offer suggestions to the user for correcting the field value via a pop-up window or some other notification means. [0021]. Hence, Kovacs teaches displaying human-readable error messages corresponding to each error object in a message area as well as offering

suggestions for correcting the field value via a pop-up window or some other notification means.

There is no language in Kovacs (and in particular paragraph [0021]) that makes any suggestion to: (1) open the preselected output archive file; and (2) tag each deployable software component in the input archive file if the step of opening the preselected output archive fails (missing claim limitations) in order to identify the input errors and offer the suggesting solution to those errors (Examiner's reasoning). The Examiner has to provide some rational connection between the cited passage that is the source of the Examiner's reasoning and the above-cited missing claim limitations. The Examiner's source of reasoning (paragraph [0021] of Kovacs) does not provide reasons as to why one skilled in the art would modify Garms to include the above-cited missing claim limitations of claims 5, 25 and 32. Accordingly, the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 5-6, 25-26 and 32-33. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007).

Further, the Examiner's reasoning ("to identify the input errors and offer the suggesting solution to those errors") does not provide reasons, as discussed further below, that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would modify Garms to include the above-indicated missing claim limitations of claims 5, 25 and 32. Accordingly, the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 5-6, 25-26 and 32-33. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007); *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998).

Garms addresses the problem of long delays in the development cycle in developing software. Column 1, line 30 – column 3, line 40. The Examiner has not provided any reasons as to why one skilled in the art would modify Garms (which addresses the problem of long delays in the development cycle in developing software)

to: (1) open the preselected output archive file; and (2) tag each deployable software component in the input archive file if the step of opening the preselected output archive fails (missing claim limitations). The Examiner's rationale ("to identify the input errors and offer the suggesting solution to those errors") does not provide such reasoning.

Why would the reason to modify Garms (whose purpose is to address the problem of long delays in the development cycle in developing software) to: (1) open the preselected output archive file; and (2) tag each deployable software component in the input archive file if the step of opening the preselected output archive fails (missing claim limitations) be to identify the input errors and offer the suggesting solution to those errors?

Garms is not concerned with identifying the input errors and offering the suggesting solution to those errors. The Examiner cannot completely ignore the teachings of Garms in concluding it would have been obvious to modify Garms to include the above-cited missing claim limitations of claims 5, 25 and 32.

Further, what is the rational connection between opening the preselected output archive file (missing claim limitation) and identifying the input errors and offering the suggesting solution to those errors (Examiner's reasoning)? Further, what is the rational connection between tagging each deployable software component in the input archive file if the step of opening the preselected output archive fails (missing claim limitation) and identifying the input errors and offering the suggesting solution to those errors (Examiner's reasoning)?

Hence, the Examiner's rationale does not provide reasons that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would modify Garms to include the above-cited missing claim limitations of claims 5, 25 and 32. Accordingly, the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 5-6, 25-26 and 32-33. *KSR International Co. v. Teleflex*

Inc., 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007); *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998).

3. Examiner fails to provide a rational underpinning for modifying Garms with Kovacs to include the missing claim limitation of claim 6.

As stated above, most if not all inventions arise from a combination of old elements. *See In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Obviousness is determined from the vantage point of a hypothetical person having ordinary skill in the art to which the patent pertains. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1457 (Fed. Cir. 1998). Therefore, an Examiner may often find every element of a claimed invention in the prior art. *Id.* However, identification in the prior art of each individual part claimed is insufficient to defeat patentability of the whole claimed invention. *See Id.* In order to establish a *prima facie* case of obviousness, the Examiner must show reasons that the skilled artisan, confronted with the same problems as the inventor and with no knowledge of the claimed invention, would select the elements from the cited prior art references for combination in the manner claimed. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1458 (Fed. Cir. 1998). The Examiner must provide articulated reasoning with some rational underpinning to support the legal conclusion of obviousness. *In re Kahn*, 441 F.3d 977, 988 (Fed. Cir. 2006) (cited approvingly in *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007)).

The Examiner admits that Garms does not teach the claim limitation of claims 6, 26 and 33. Office Action (12/22/2008), page 10. The Examiner asserts that Kovacs teaches the limitation of claim 6. *Id.* However, the Examiner has not provided any rational underpinning for modifying Garms with Kovacs to include this missing claim limitation. Hence, the Examiner has not provided a *prima facie* case of obviousness in rejecting claims 6, 26 and 33. *KSR International Co. v. Teleflex Inc.*, 82 U.S.P.Q.2d 1385, 1396 (U.S. 2007); M.P.E.P. §2143.

III. CONCLUSION:

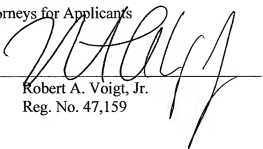
As a result of the foregoing, it is asserted by Applicants that claims 1-7 and 21-33 in the Application are in condition for allowance, and Applicants respectfully request an allowance of such claims. Applicants respectfully request that the Examiner call Applicants' attorney at the below listed number if the Examiner believes that such a discussion would be helpful in resolving any remaining issues.

Respectfully submitted,

WINSTEAD P.C.

Attorneys for Applicants

By: _____


Robert A. Voigt, Jr.
Reg. No. 47,159

P.O. Box 50784
Dallas, TX 75201
(512) 370-2832